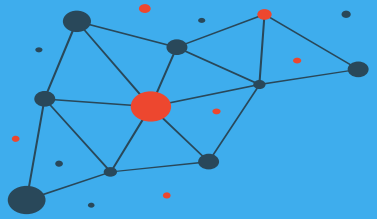
Spectrum Scale Expert Talks

Episode 17:

Spectrum Scale Network Performance Enhancement Overview (MCOT)

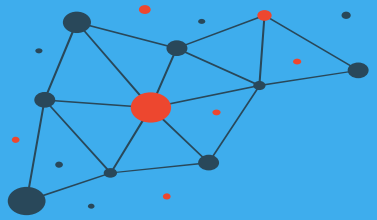
Show notes:
www.spectrumscaleug.org/experttalks

Join our conversation:
www.spectrumscaleug.org/join



About the user group

- Independent, work with IBM to develop events
- Not a replacement for PMR!
- Email and Slack community
- https://www.spectrumscaleug.org/join



We are …

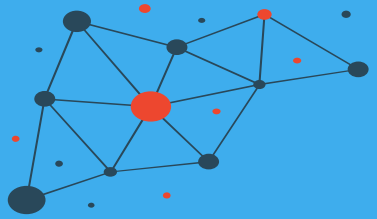
Current User Group Leads

- Paul Tomlinson (UK)
- Kristy Kallback-Rose (USA)
- Bob Oesterlin (USA)

Former User Group Leads

- Simon Thompson (UK)
- Bill Anderson (USA)
- Chris Schlipalius (Australia)

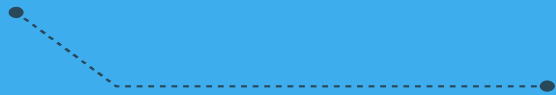
IBM CHAMPION

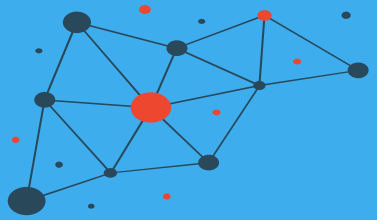


Check https://www.spectrumscaleug.org/experttalks for charts, show notes and upcoming talks

- Past talks:
  - 001: What is new in Spectrum Scale 5.0.5?
  - 002: Best practices for building a stretched cluster
  - 003: Strategy update
  - 004: Update on performance enhancements in Spectrum Scale (file create, MMAP, direct IO, ESS 5000)
  - 005: Update on functional enhancements in Spectrum Scale (inode management, vCPU scaling, NUMA considerations)
  - 006: Persistent Storage for Kubernetes and OpenShift environments
  - 007: Manage the lifecycle of your files using the policy engine
  - 008: Multi-node scaling of AI workloads using Nvidia DGX, OpenShift and Spectrum Scale
  - 009: Continental: Deep Thought – An AI Project for Autonomous Driving Development
  - 010: Data Accelerator for Analytics and AI (DAAA)
  - 011: What is new in Spectrum Scale 5.1.0?
  - 012: Lenovo - Spectrum Scale and NVMe Storage
  - 013:Event driven data management and security using Spectrum Scale Clustered Watch Folder and File Audit Logging
  - 014: What is new in Spectrum Scale 5.1.1?
  - 015: IBM Spectrum Scale Container Native Storage Access
  - 016: What is new in Spectrum Scale 5.1.2?

  This talk
  - 017: Multiple Connections over TCP (MCOT)





Speakers

- Jay Vaddi

- Loads of thanks to John Lewars and Yong Ze Chen!

Disclaimer

IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM's sole discretion. Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision. The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code, or functionality. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

IBM reserves the right to change product specifications and offerings at any time without notice. This publication could include technical inaccuracies or typographical errors. References herein to IBM products and services do not imply that IBM intends to make them available in all countries.

IBM’s statements regarding its plans, directions, and intent are subject to change or withdrawal without notice and at IBM’s sole discretion.

Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract.

The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user’s job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

Scale 5.1.0 and Older Versions (Single Connection over TCP)

> When TCP/IP is used by Spectrum Scale, only one TCP connection is used to send data between any given pair of nodes.

> **Non-Bonded Config**: A single TCP connection will always use a single physical interface to send the packets.

> **Bonded Config**: With the preferred link aggregation mode like mode 4 (802.3ad or LACP aggregation), only a single interface is used.
- It's possible to use a link aggregation mode like round robin (mode=1) that will utilize all the available links, but this mode is generally discouraged because of possibility of out of order packets.

> A single TCP connection often can't saturate the network bandwidth[1].

1. Depends on several factors such as tuning, CPU memory copy rates and network adapter speed.

Scale 5.1.1 (Multiple Connections over TCP - MCOT)

> Scale can create multiple TCP connections to the same destination IP address.

> **Non-Bonded Config:** More TCP connections on single interface allow for better utilization of network bandwidth[1].

> **Bonded Config:** More TCP connections on a bonded interface (mode 4) allow for use of more physical interfaces and improves bonding balance issues[2].

>The number of connections can be controlled by Scale configuration option *maxTcpConnsPerNodeConn*. It can be changed using *mmchconfig* command.

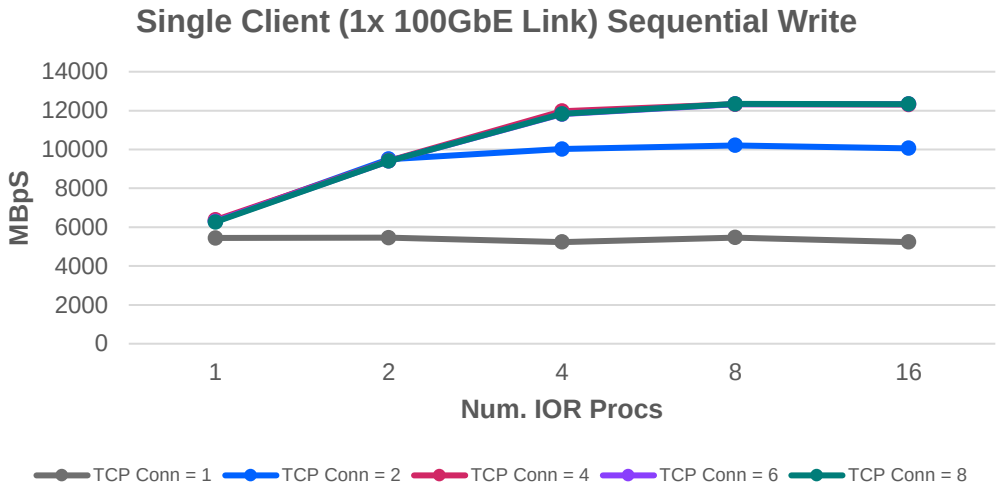
1. Depends on several factors such as tuning, CPU memory copy rates and network adapter speed.
2. Depends on mode 4 xmit_hash_policy used and load balancing algorithm on the switch.

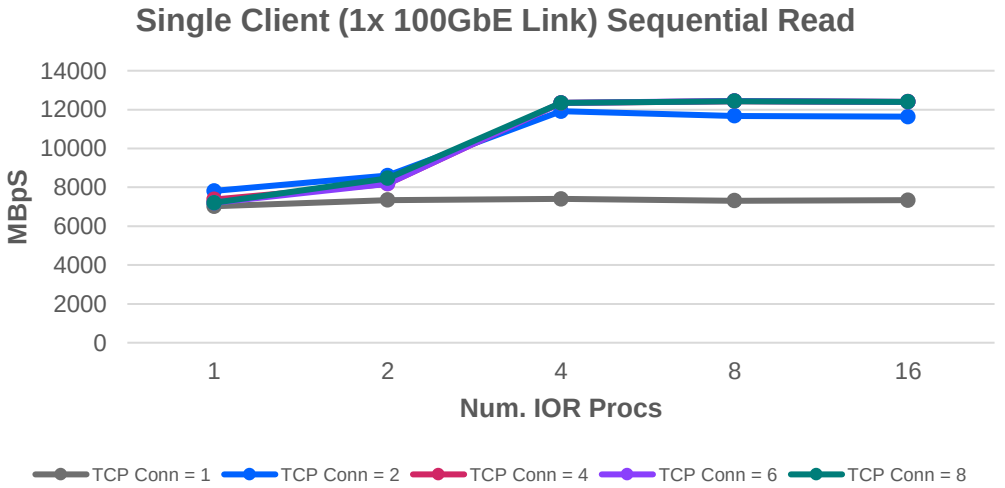
Single Client Sequential Bandwidth

> Single 100 Gb Ethernet link can deliver up to 12.5 GB/s (theoretical).

> An example measurement done in IBM Lab:
- A client node, with 1x 100GbE link, connected to IBM Elastic Storage System (ESS).
- With single TCP connection the bandwidth was only ~5 GB/s for write and ~7 GB/s for read. But, with multiple TCP connections the client network bandwidth was saturated.

> The single TCP connection performance and the improvement with multiple connections depend on the hardware capability.

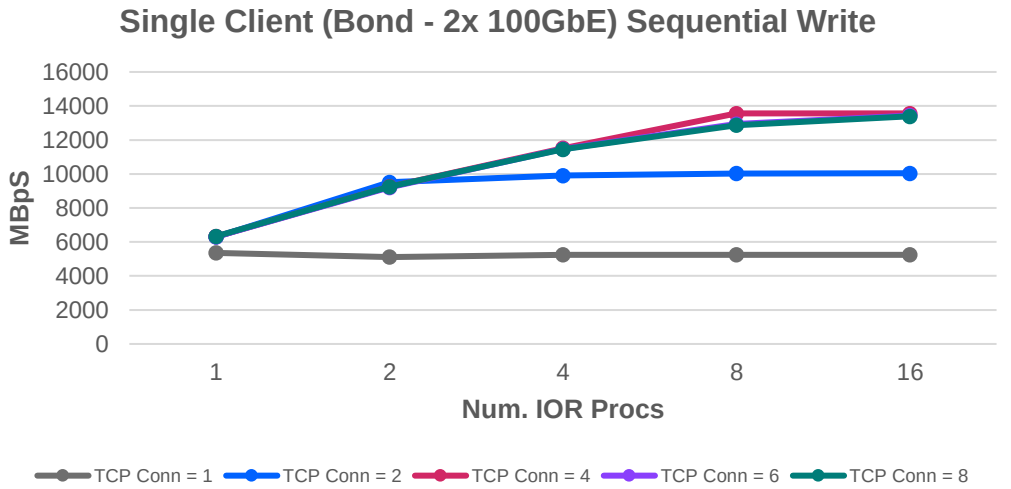


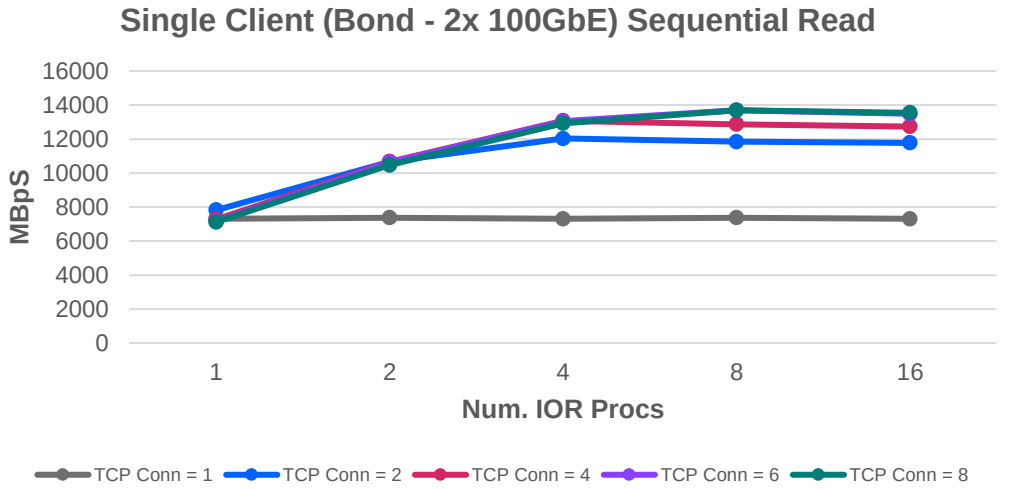


Single Client Sequential Bandwidth (cont.)

> 2x 100GbE links bonded (mode=4, xmit_hash_policy=3+4) can deliver up to 25 GB/s (theoretical) with multiple TCP connections.

> An example measurement done in IBM Lab:
- A client node, with 2x 100GbE link (bonded, mode=4, xmit_hash_policy=3+4), connected to IBM Elastic Storage System (ESS).
- With extensive tuning experiments we're limited to no more than ~14 GB/s with multiple connections, which we believe is a results of system fabric bottleneck on the client node, preventing it from achieving the theoretical network bandwidth.

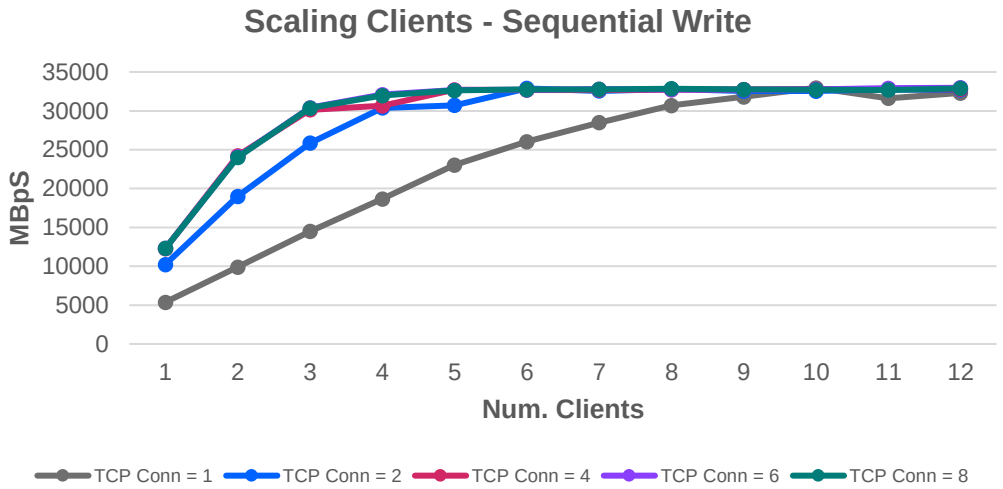


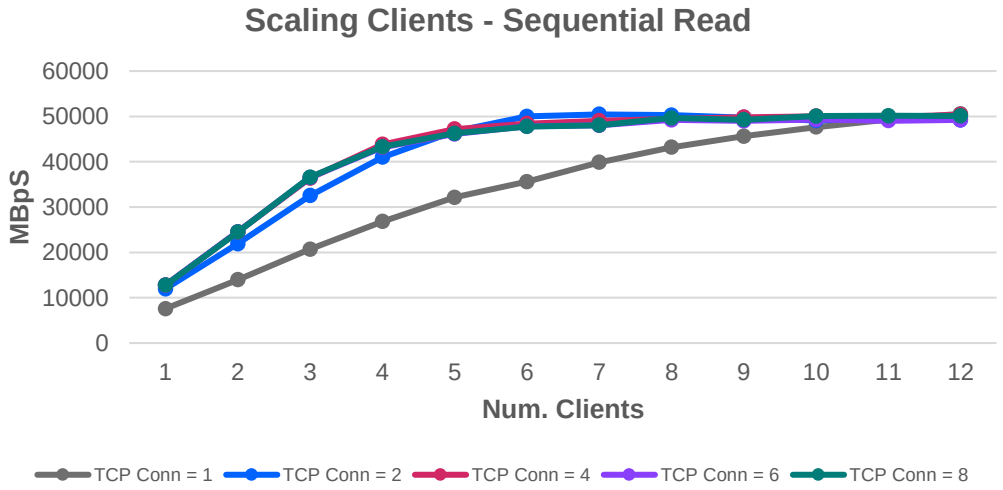


Multiple Clients Sequential Bandwidth

> An example measurement done in IBM lab:
- Client nodes, each with 1x 100GbE link, connected to IBM Elastic Storage System (ESS).
- With multiple connections the full TCP/IP bandwidth of ESS can be achieved with fewer client nodes[1].

> There are some cases in which a client node, and not the NSD server, may be the bottleneck for a given set of clients' maximum read or write bandwidth (for example, in HDFS configurations data flows through the HDFS data node).





1. Depends on several factors such as tuning, CPU memory copy rates and network bandwidth available per Scale client node.

Scale TCP Connections Setting

> maxTcpConnsPerNodeConn controls the maximum number of TCP connections that the GPFS daemon will establish to another node. Valid values are 1-8, with the default being 2.
- A GPFS daemon restart is required for the new maxTcpConnsPerNodeConn value to take effect.

> The value of maxTcpConnsPerNodeConn can be tuned according to the size of cluster, overall network bandwidth, and maxReceiverThreads. But the important consideration is the size of cluster. For example, a cluster with 500 nodes can generally achieve the use of all interfaces and a fair balance of TCP traffic across interfaces with one TCP connection. While a cluster with 5 nodes needs multiple TCP connections to achieve concurrent use of interfaces and have fair balance.

> The number of TCP connections between a pair of Scale nodes will be the minimum of the configured value of maxTcpConnsPerNodeConn on those nodes.

Bonding - 802.3ad or LACP Aggregation

> 802.3ad or LACP Aggregation is the most used link aggregation mode for Scale deployments.

> 802.3ad or LACP Aggregation determines the interface to use based on hash of packet's source and destination information.

> LACP2+3 (xmit_hash_policy=2 or layer2+3) - uses the IP and MAC info of source and destination to generate hash. Even with multiple connections the hash remains same.

(((source_IP XOR dest_IP) AND 0xffff) XOR (source_MAC XOR destination_MAC)) MODULO slave_count

> LACP3+4 (xmit_hash_policy=1 or layer3+4) - uses the IP and Port info of source and destination to generate hash. With multiple connections, multiple TCP port numbers are used and there is a better chance of multiple interfaces being used.

((source_port XOR dest_port) XOR ((source_IP XOR dest_IP) AND 0xffff) MODULO slave_count

>The load balancing algorithm on the switch is important to ensure better balancing across links from switch to destination.

Thank you!

Please help us to improve Spectrum Scale with your feedback

- If you get a survey in email or a popup from the GUI, please respond
- We read every single reply

Spectrum Scale User Group

The Spectrum Scale (GPFS) User Group is free to join and open to all using, interested in using or integrating IBM Spectrum Scale.

The format of the group is as a web community with events held during the year, hosted by our members or by IBM.

See our web page for upcoming events and presentations of past events. Join our conversation via mail and Slack.

www.spectrumscaleug.org